

AMENDMENTS TO THE CLAIMS

1. (Currently amended) A block-level shared network storage system, comprising:
a storage server comprising an array of disk drives, and comprising a processor that runs a device driver to provide block-level access to data stored on the array of disk drives, wherein the storage server is configurable to provide multiple storage partitions, each of which may be allocated to a different host computer; and
a host computer coupled to the storage server by at least one computer network;
wherein the host computer and the storage server perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections, each logical connection being between the host computer and the storage server over the at least one computer network, such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection.
2. (Original) The network storage system as in Claim 1, wherein each logical connection is a socket connection.
3. (Original) The network storage system as in Claim 2, wherein each socket connection is a TCP/IP socket connection.
4. (Original) The network storage system as in Claim 1, wherein each logical connection remains persistent over multiple I/O operations performed over that logical connection.
5. (Canceled)
6. (Previously presented) The network storage system as in Claim 1, wherein the storage server provides virtualized block-level storage access to the host computer such that the storage server is treated as local disk drive storage by user-level processes running on the host computer.
7. (Canceled)
8. (Original) The network storage system as in Claim 1, wherein the storage server has a first storage partition which is uniquely assigned to the host computer such that the first

storage partition appears to user-level processes running on the host computer as a private, local disk drive.

9. (Original) The network storage system as in Claim 8, wherein the storage server further has a second storage partition which is uniquely assigned to a second host computer.

10. (Original) The network storage system as in Claim 8, wherein the storage server further has a second storage partition which is shared by multiple host computers.

11. (Original) The network storage system as in Claim 1, wherein the host computer and the storage server implement an authentication protocol in which the storage server authenticates the host computer before allowing the host computer to perform input/output operations.

12. (Original) The network storage system as in Claim 1, wherein the host computer and the storage server implement a discovery protocol in which the storage server notifies the host computer of partitions assigned to the host computer.

13. (Original) The network storage system as in Claim 1, wherein at least one of the logical connections is over a general-purpose computer network.

14. (Original) The network storage system as in Claim 1, wherein at least one of the logical connections is over an Ethernet network.

15. (Original) The network storage system as in Claim 1, wherein the first and second logical connections exist over separate computer networks.

16. (Original) The network storage system as in Claim 1, wherein each logical connection exists between a respective reader/writer pair.

17. (Original) The network storage system as in Claim 1, wherein the host computer and the storage server are interconnected by at least one switch.

18. (Canceled)

19. (Currently amended) A system for storing data for host computers, comprising:

a plurality of storage servers connected to a network, each storage server comprising an array of disk drives, an array controller, and a processor;

a plurality of host computers connected to the network and programmed to store data on the storage servers; and

at least one switch which interconnects the plurality of storage servers with the plurality of host computers;

wherein each host computer is programmed to open multiple concurrent socket connections over the network to the storage servers and to perform input/output operations in parallel over the multiple concurrent socket connections;

wherein a first storage server of the plurality of storage servers has a first partition which is uniquely assigned to a first host computer of the plurality of host computers such that the first partition appears as a local disk drive to the first host computer.

20. (Previously presented) The system of Claim 19, wherein the socket connections are TCP/IP socket connections.

21. (Previously presented) The system of Claim 19, wherein each storage server of the plurality of storage servers provides virtualized block-level storage access to the host computers such that the storage servers are treated as local disk drive storage by user-level processes running on the host computers.

22. (Canceled)

23. (Original) The system of Claim 19, wherein a first storage server of the plurality of storage servers is configurable to provide multiple, variable-size partitions, each of which may be allocated to a different host computer of the plurality of host computers.

24. (Canceled)

25. (Currently amended) The system of ~~Claim 24~~ Claim 19, wherein the first storage server further has a second partition which is uniquely assigned to a second host computer of the plurality of host computers.

26. (Original) The system of Claim 19, wherein the host computers and the storage servers implement an authentication protocol in which a storage server authenticates a host computer before allowing the host computer to perform input/output operations.

27. (Previously presented) The system of Claim 19, wherein the host computers and the storage servers implement a discovery protocol in which a storage server notifies a host computer of partitions assigned to the host computer.

28. (Previously presented) A method of performing input/output operations, comprising:

establishing first and second TCP/IP connections between a host computer and a block-level storage server over one or more computer networks;

performing a first input/output operation over the first TCP/IP connection concurrently with performing a second input/output operation over the second TCP/IP connection, each of said input/output operations comprising a transfer of input/output data between the host computer and the storage server; and

maintaining the first and second TCP/IP connections in a persistent state such that each TCP/IP connection may be used to perform additional input/output operations.

29. (Original) The method as in Claim 28, wherein the first and second TCP/IP connections are established over separate computer networks.

30. (Original) The method as in Claim 28, wherein the first input/output operation is a first I/O request issued from a first application running on the host computer, and the second input/output operation is a second I/O request issued from a second application running on the host computer.

31. (Original) The method as in Claim 28, wherein the first and second input/output operations are constituent operations of an I/O request issued by a process running on the host computer, whereby the I/O request is executed in parallel over multiple TCP/IP connections.

32. (Original) The method as in Claim 28, further comprising establishing a third TCP/IP connection between the host computer and the storage server, and using the third TCP/IP connection to perform an authentication sequence in which the storage server authenticates the host computer.

33. (Original) The method as in Claim 32, further comprising conveying access information over the third TCP/IP connection from the storage server to the host computer, said access information specifying access rights uniquely assigned to the host computer.

34. (Original) A method of executing an input/output (I/O) request received from a user-level process running on a host computer, comprising:

on the host computer, dividing the I/O request into multiple constituent I/O operations; and

performing the multiple constituent I/O operations in parallel over multiple, respective logical network connections between the host computer and a target storage server such that I/O data is transferred between the host computer and the storage server over each of the logical network connections.

35. (Original) The method of Claim 34, wherein each logical network connection is a socket connection.

36. (Original) The method of Claim 34, wherein each logical network connection is a TCP/IP socket connection.

37. (Original) The method of Claim 34, wherein at least one of the logical network connections is over a general-purpose computer network.

38-53: (Canceled)

54. (Previously presented) The network storage system as in Claim 1, wherein the host computer and the storage server communicate with each other over each of the logical connections using a TCP/IP protocol.

55. (Previously presented) The system of Claim 19, wherein a first host computer of said plurality of host computers is programmed to open first and second socket connections over the network to a first storage server of said plurality of storage servers, and to perform a first input/output operation over the first socket connection while performing a second input/output operation over the second socket connection.

56. (Previously presented) The system of Claim 19, wherein the host computers and storage servers are programmed to perform said input/output operations via TCP/IP communications over said socket connections.

57. (Previously presented) The method of Claim 28, wherein the method comprises establishing and maintaining the first and second TCP/IP connections over an Ethernet network.

58. (Previously presented) The method of Claim 34, wherein performing the multiple constituent I/O operations comprises the host computer communicating with the target storage server over the multiple logical network connections using a TCP/IP protocol.

59. (Previously presented) The method of Claim 34, wherein the multiple constituent I/O operations are performed over an Ethernet network to which the host computer and the storage server are connected.

60. (New) A block-level shared network storage system, comprising:

a storage server comprising an array of disk drives, and comprising a processor that runs a device driver to provide block-level access to data stored on the array of disk drives; and

a host computer coupled to the storage server by at least one computer network;

wherein the host computer and the storage server perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections, each logical connection being between the host computer and the storage server over the at least one computer network, such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection, and the host computer is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical connections of said multiple, concurrent logical connections

61. (New) A system for storing data for host computers, comprising:

a plurality of storage servers connected to a network, each storage server comprising an array of disk drives, an array controller, and a processor;

a plurality of host computers connected to the network and programmed to store data on the storage servers; and

at least one switch which interconnects the plurality of storage servers with the plurality of host computers;

wherein each host computer is programmed to open multiple concurrent socket connections over the network to the storage servers and to perform input/output

operations in parallel over the multiple concurrent socket connections, and at least a first host computer of the plurality of host computers is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical socket connections between the first host computer and a target storage server.

62. (New) A block-level shared network storage system, comprising:

a storage server comprising an array of disk drives, and comprising a processor that runs a device driver to provide block-level access to data stored on the array of disk drives; and

a host computer coupled to the storage server by at least one computer network;

wherein the host computer and the storage server perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections, each logical connection being between the host computer and the storage server over the at least one computer network, such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection, and the host computer and the storage server each include two network interfaces that provide redundant network connections between the host computer and the storage server.